**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
ON APPEAL FROM THE EXAMINER TO THE BOARD
OF PATENT APPEALS AND INTERFERENCES**

In re Application of:	HARROW, et al.
Serial No.:	09/465,236
Filing Date:	December 15, 1999
Group No.:	2661
Examiner:	Ricardo M. Pizarro
Confirmation No.:	8644
Title:	SYSTEM AND METHOD FOR USING A PLURALITY OF PROCESSORS TO SUPPORT A MEDIA CONFERENCE

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Appeal Brief

Appellants have appealed to the Board of Patent Appeals and Interferences ("Board") from the decision of the Examiner mailed February 23, 2006, finally rejecting pending Claims 2-7, 9-14, and 32-40. Appellants filed a Notice of Appeal on March 10, 2006, along with a one-month Extension of Time. Appellants respectfully submit this Appeal Brief with the statutory fee of $500.00.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The real party in interest for this Application under appeal is Cisco Technology, Inc. of San Jose, California.

This application is currently owned by Cisco Technology, Inc., as indicated by an assignment recorded on December 15, 1999 in the Assignment Records of the United States Patent and Trademark Office at Reel 010451, Frame 0575.

RELATED APPEALS AND INTERFERENCES

The Appellant, the undersigned Attorney for Appellant, and the Assignee know of no applications on appeal that may directly affect or be directly affected by or have a bearing on the Board's decision in this pending appeal.

STATUS OF CLAIMS

All pending claims—Claims 2-7, 9-14, and 32-40—were rejected by the Final Office Action dated September 12, 2005 and Advisory Action dated February 23, 2006, and are presented for appeal, as shown in Appendix A.

STATUS OF AMENDMENTS

The claims on appeal, which appear in Appendix A, represent the form of the claims as of the time of the Final Office Action dated September 12, 2005 and Advisory Action dated February 23, 2006. Appellants filed no amendments to the claims after the Final Office Action.

SUMMARY OF CLAIMED SUBJECT MATTER

The claims of the present application are directed to methods and systems for using a plurality of processors to support a media conference call. A media conference is a real-time or near real-time communication among three or more participants. Specification, p. 6, ll. 3-4. To establish and maintain media conferences a system (2) includes a data network (4), end-user devices (6a, 6b, and 6c), a gateway device (8), and a conferencing device (10). *Id.* at p. 7, ll. 4-6. Using media transformation processors and mixing processors, the conferencing device allows participants in a media conference to share media information in a real-time or near real-time environment. *Id.* at p. 7, ll. 6-9.

To support a media conference, the conferencing device performs three basic operation. *Id.* at p. 7, ll. 6-7. First, the conferencing device receives input data streams from participants' end-user devices and decodes the input data streams to generate input media information. *Id.* at p. 7, ll. 7-9. Second, the conferencing device mixes the input media information to generate output media information. *Id.* at p. 7, ll. 9-10. Third, the conferencing device encodes the output media information to generate output data streams and communicates the output data streams to participants' end-user devices. *Id.* at p. 7, ll. 10-12.

To perform these operations, the conferencing device includes media transformation processors (12a, 12b, and 12c), mixing processors (14a, 14b, and 14c), and a system resource management (SRM) module (16). *Id.* at p. 8, ll. 22-25. The SRM module assigns the various decoding, mixing, and encoding operations to the media transformation processors and mixing processor. *Id.* at p. 8, ll. 25-27. By assigning decoding or encoding operations to the media transformation processors, SRM module relieves the burden of supporting a media conference from a single mixing processor. *Id.* at p. 8, ll. 27-29. With this multi-processor solution, the conferencing device avoids restricting the size of a media conference based on the limited resources of any single processor. *Id.* at p. 8, ll. 29-31. In addition, the conferencing device may devote greater resources to a media conference and, as a result, support more processing-intensive coding standards that facilitate communication of data streams over the data network. *Id.* at p. 8, l. 31 - p. 9, l. 1.

The media transformation processors and mixing processors represent separate hardware components. *Id.* at p. 9, ll. 3-4. The functionality described below may be implemented using separate hardware components or software that executes using the

separate hardware components. *Id.* at p. 9, ll. 4-6. Thus, a media transformation processor and a mixing processor do not operate using the same actual physical computing machinery. *Id.* at p. 9, ll. 6-7. The media transformation processors and mixing processors may represent separate microprocessors, controllers, digital signal processors (DSPs), or other integrated circuit chips mounted to a circuit board. *Id.* at p. 9, ll. 8-10. Alternatively, the media transformation processors and mixing processors may represent separate networks of electronic components, such as transistors, diodes, resistors, etc., and their interconnections etched or imprinted on a single chip. *Id.* at p. 9, ll. 10-13. In such an embodiment, the media transformation processors and the mixing processors may use shared resources but generally rely on separate pipelines to perform the majority of their processing. *Id.* at p. 9, ll. 13-15. Although the media transformation processors and the mixing processors represent separate hardware components, the hardware components are not necessarily different in type. *Id.* at p. 9, ll. 15-17.

The media transformation processors may receive input data streams from participants' end-user devices, decode the input data streams to generate input media information, and communicate the input media information to mixing processors. *Id.* at p. 9, ll. 20-23. The SRM module assigns a participant's input data stream to a media transformation processor and communicates data packets associated with the participant to the media transformation processor. *Id.* at p. 9, ll. 23-25. Using the data packets, the media transformation processor reconstructs the input data stream generated by the participant's end-user device. *Id.* at p. 9, ll. 25-27. Specifically, media transformation processor 12 may resequence the received data packets, insert replacement data packets for any missing data packets, or otherwise rehabilitate the media stream. *Id.* at p. 9, ll. 27-29. After reconstructing the input data stream from the data packets, the media transformation processor decodes the input data stream to generate input media information. *Id.* at p. 10, l. 30 - p. 11, l. 1.

In addition to decoding input data streams, the media transformation processors may also receive output media information from mixing processors, encode the output media information to generate output data streams, and communicate the output data streams to participants' end-user devices. *Id.* at p. 11, ll. 12-15. The SRM module assigns an output data stream to media transformation processor, and in response, the media transformation processor receives from the mixing processor output media information associated with the participant. *Id.* at p. 11, ll. 15-18. The media transformation processor encodes the output

media information for communication over the data network. *Id.* at p. 11, ll. 18-19. By encoding the output media information, the media transformation processor may compress the output media information into fewer number of bits to facilitate communication over the data network. *Id.* at p. 11, ll. 19-22. In a voice telephone conference, the media transformation processor may encode input voice information according to G.711, G.723, G.729, or any other voice coding or compression standard. *Id.* at p. 11, ll. 22-24. After encoding the output media information, the media transformation processor communicates the resulting output data stream to the participant's end-user device 6 using data network 4. *Id.* at p. 12, ll. 9-11.

The mixing processor mixes input media information to generate output media information. *Id.* at p. 12, ll. 21-22. The mixing processor may receive input media information from media transformation processors. *Id.* at p. 12, ll. 22-23. Alternatively, like media transformation processors, the mixing processor may receive input data streams from participants' end-user devices and decode the input data streams to generate input media information. *Id.* at p. 12, ll. 23-25. The mixing processor mixes the input media information associated with two or more participants to generate output media information. *Id.* at p. 12, ll. 26-27. For example, the mixing processor may mix input voice information from two or more participants to generate output voice information. *Id.* at p. 12, ll. 27-29. If the media conference includes video information, the mixing processor may synchronize the video information with the output voice information. *Id.* at p. 12, ll. 29-31. Similarly, the mixing processor may also enable participants to share other media information. *Id.* at p. 13, ll. 2-3. For example, when sharing a spreadsheet, a word processing document, a whiteboard presentation, or other software application information, the mixing processor copies the input media information to generate output media information for each of the conference participants. *Id.* at p. 13, ll. 3-7. After generating the output media information, the mixing processor may communicate the output media information to a media transformation processor for encoding and communication to conference participants. *Id.* at p. 13, ll. 7-9. Alternatively, like media transformation processors, the mixing processor may encode the output media information to generate output data streams and communicate the output data streams to participants' end-user devices. *Id.* at p. 13, ll. 9-12.

The SRM module allocates media conferences to media transformation processors and mixing processors and, accordingly, stores status information relating to the media

conferences in memory (18). *Id.* at p. 13, ll. 22-24. The SRM module may be implemented using hardware, software stored in a computer readable medium, or a combination of both hardware and software. *Id.* at p. 13, ll. 24-26. When conference participants create a new media conference, the SRM module receives initiation information indicating a number of participants in the media conference. *Id.* at p. 13, ll. 28-20. The SRM module selects one of mixing processors to support the media conference. *Id.* at p. 13, ll. 30-31. As described above, the mixing processor mixes input media information associated with the conference participants to generate output media information. *Id.* at p. 13, ll. 31 - p. 14, l. 2. The SRM module may also assign the selected mixing processor the tasks of decoding input data streams to generate the input media information and encoding the output media information to generate output data streams. *Id.* at p. 14, ll. 2-4. Alternatively, to provide multi-processor support for the media conference, the SRM module assigns some or all of the decoding and encoding operations to media transformation processors. *Id.* at p. 14, ll. 4-7. The SRM module may assign the task of decoding a participant's input data stream and the task of encoding output media information for communication to the participant to the same or different media transformation processors. *Id.* at p. 14, ll. 7-9. In response to assigning decoding, mixing, and encoding tasks to media transformation processors and mixing processors, the SRM module stores status information relating to the media conference in the memory. *Id.* at p. 14, ll. 13-15. To control communication among media transformation processors and mixing processors, the SRM module communicates control information to media transformation processors and mixing processors. *Id.* at p. 14, ll. 23-55.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Appellants request that the Board review the Examiner's rejection of Claims 38, 2-4, 7, 39, 9-11, and 14 under 35 U.S.C. § 102(a) as being anticipated by U.S. Patent No. 6,463,414 ("*Su*").

II. Appellants request that the Board review the Examiner's rejection of Claims 40, 32-34, and 37 under 35 U.S.C. § 102(a) as being anticipated by *Su*.

III. Appellants request that the Board review the Examiner's rejection of Claims 5, 6, 12, 13, 35, and 36 under 35 U.S.C. § 103(a) as being unpatentable by *Su* in view of U.S. Patent No. 5,841,763 ("*Leondires*").

ARGUMENT

I. SU FAILS TO DESCRIBE, EXPRESSLY OR INHERENTLY, THE INVENTIONS RECITED IN CLAIMS 38, 2-4, 7, 39, 9-11, AND 14.

A. Su is Not Prior Art Under 35 U.S.C. § 102(a) as Asserted by the Examiner.

In the Final Office Action mailed September 12, 2005, the Examiner erroneously cited *Su* at prior art under 35 U.S.C. § 102(a). (Final Office Action at p. 2.) *Su* issued on October 8, 2002, which is after Applicants' December 15, 1999 filing date. Thus, *Su* is not prior art under 35 U.S.C. § 102(a).

Furthermore, *Su* is not prior art under § 102(e) because it was filed on April 12, 2000, which is almost four months ***after*** Applicants' filing date of December 15, 1999. While *Su* claims priority to a provisional application filed on April 12, 1999, that provisional application does not support the entire disclosure of *Su*. At most, *Su* may qualify as prior art only to the extent the disclosure is supported by Provision Application No. 09/547,832 ("*Su Provisional Application*"). A copy of the *Su Provisional Application* is submitted in Appendix B.

B. The Su Provisional Application Does Not Disclose Several Elements Required by Claims 38, 2-4, 7, 39, 9-11, and 14.

Appellants' Claim 38 recites:

> An apparatus for using a plurality of processors to support a media conference, comprising:
>
> a mixing processor operable to mix input media information associated with two or more first participants to generate output media information for communication to a second participant; and
>
> a first media transformation processor coupled to the mixing processor, the first media transformation processor operable to receive the output media information from the mixing processor, to encode the output media information to generate an output data stream, and to communicate the output data stream to the second participant's end-user device,
>
> wherein the mixing processor and the first media transformation processor are separate hardware components.

Appellants respectfully submit that the *Su Provisional Application* fails to describe, expressly or inherently, every element of this claim, and therefore the Examiner's § 102 rejection based on the *Su Provisional Application* must fail. *See In re Robertson*, 169 F.3d 743, 745, 49 U.S.P.Q.2d 1949, 1950 (Fed. Cir. 1999) (stating that a single prior art reference must describe, either expressly or inherently, each and every element of the claim in order to anticipate a claim under 35 U.S.C. § 102(e)).

Among other elements, the *Su Provisional Application* does not disclose, teach, or suggest the use of separate processors, such as the mixing processor and media transformation processor of Claim 38. Independent Claims 38 requires a "mixing processor" and a "first media transformation processor" and, as stated in the claim, "the mixing processor and the first media transformation processor are separate hardware components." The Examiner's rejections are based on Figures 2 and 5 of *Su*, but these figures are not prior art because they are not included in the *Su Provisional Application* (which itself may or may not be prior art). The *Su Provisional Application* does not describe or illustrate the use of separate processors as recited in Claim 38. At most, the *Su Provisional Application* shows functional blocks (as in Figure 2) as opposed to separate hardware components.

Moreover, even if one were to consider the more detailed disclosure of *Su* (which does ***not*** qualify as prior art), it still does not disclose separate hardware components. The Examiner has several times acknowledged that *Su* does not disclose separate processors. In the Office Action mailed March 1, 2005 and the Final Office Action mailed September 12, 2005, the Examiner stated:

> ***Su did not specifically disclose said processors being separate as in claims 5, 12, and 35***, being DSP as in claim 6, 13 and 36.

(p. 6) (emphasis added). While the Examiner later stated in the Final Office that this statement is referring only to the DSP processors, that is plainly not true. The Examiner has stated that, in addition to not disclosing DSP as in Claims 6, 13, and 36, *Su* also does not disclose the "processors being separate as in claims 5, 12, and 35."

Furthermore, the written description of *Su* confirms that *Su* does not disclose the "processors being separate as in claims 5, 12, and 35" as stated by the Examiner. *Su* expressly states that the invention is described in terms of "functional block components" which may be implemented using "any number of hardware components or software elements":

> The present invention may be described herein in terms of ***functional block components*** and various processing steps. It should be appreciated that such functional blocks may be realized by ***any number of hardware components or software elements*** configured to perform the specified functions. For example, the present invention may employ various integrated circuit components, e.g., memory elements, digital signal processing elements, logic elements, look-up tables, and the like, which may carry out a variety of functions under the control of one or more microprocessors or other control devices.

(Col. 2, ll. 49-59) (emphasis added). *Su* does not specify that the functions of decoders 230 and 234, mixer 238 and 240, and encoder 232 and 236 are assigned to separate processors. Indeed, *Su* provides that the functional blocks may be implemented in software.

Another passage in *Su* further indicates that Figure 2, on which the Examiner relies to support the rejections, is a "simplified schematic" of functional blocks as opposed to hardware components.

> FIG. 2 is a simplified schematic: there might also be certain additional components advantageously coupled between the packet network and the decoders (and encoders). Specifically, with respect to the decoders, there will likely be ***a functional block*** (not shown) that receives the packets from packet network 201 and removes all unnecessary routing, encryption, and protection information (a "decapsulator"). Conversely, with respect to the encoders, there will likely be ***a functional block*** (an "encapsulator") for each encoder that receives speech samples from the mixer and adds certain information regarding routing, encryption, and the like prior to sending the packets out over packet network 201.

(Col. 5, ll. 19-31) (emphasis added).

For at least these reasons, the *Su Provisional Application* does not disclose, teach, or suggest the "mixing processor" and "first media transformation processor," "wherein the mixing processor and the first media transformation processor are separate hardware components," as recited in Claim 38. Accordingly, Applicants respectfully request reconsideration and allowance of independent Claims 38, as well as Claims 2-7 which depend from Claim 38.

Independent Claim 39 includes limitations that, for substantially similar reasons, are not disclosed by the *Su Provisional Application*.

Because *Su* does not disclose, expressly or inherently, every element of independent Claims 38 and 39 and their respective dependent Claims 2-7 and 9-14, respectively,

Appellants respectfully request the Board to reverse the Examiner's rejection of Claims 38, 2-7, 39, and 9-14 and direct the Examiner to issue a notice of allowance.

II. Su Fails to Describe, Expressly of Inherently, the Inventions Recited in Claims 40, 32-34, and 37.

A. Su is Not Prior Art Under 35 U.S.C. § 102(a) as Asserted by the Examiner.

In the Final Office Action mailed September 12, 2005, the Examiner erroneously cited *Su* at prior art under 35 U.S.C. § 102(a). (Final Office Action at p. 4.) *Su* issued on October 8, 2002, which is after Applicants' December 15, 1999 filing date. Thus, *Su* is not prior art under 35 U.S.C. § 102(a).

Furthermore, *Su* is not prior art under § 102(e) because it was filed on April 12, 2000, which is almost four months ***after*** Applicants' filing date of December 15, 1999. While *Su* claims priority to a provisional application filed on April 12, 1999, that provisional application does not support the entire disclosure of *Su*. At most, *Su* may qualify as prior art only to the extent the disclosure is supported by Provision Application No. 09/547,832 ("*Su Provisional Application*"). A copy of the *Su Provisional Application* is submitted in Appendix B.

B. The Su Provisional Application Does Not Disclose Several Elements Required by Claims 40, 32-34, and 37.

Appellants' Claim 40 recites:

> A system for using a plurality of processors to support a media conference, comprising:
>
> a plurality of end-user devices coupled to a data network and operable to generate input media information, to encode the input media information to generate input data streams, and to communicate the input data streams using the data network; and
>
> a conferencing device coupled to the data network, the conferencing device comprising two or more processors operable to decode the input data streams to generate the input media information, to mix the input media information to generate output media information, and to encode the output media information to generate output data streams, wherein the processors are separate hardware components;

> wherein the end-user devices are further operable to receive the output data streams and to decode the output data streams to generate output media information

Among other elements, the *Su Provisional Application* does not disclose, teach, or suggest "the conferencing device comprising two or more processors operable to decode the input data streams to generate the input media information, to mix the input media information to generate output media information, and to encode the output media information to generate output data streams, wherein the processors are separate hardware components," as recited in Claim 40. Thus, like Claims 38 and 39, independent Claim 40 requires multiple processors.

The Examiner's rejections are based on Figures 2 and 5 of *Su*, but again these figures are not prior art because they are not included in the *Su Provisional Application*. Nowhere does the *Su Provisional Application* describe or illustrate the use of separate processors as recited in Claim 38. Figure 2 of the *Su Provisional Application* (which may or may not be prior art) does not portray separate processors. At most, the *Su Provisional Application* shows functional blocks as opposed to separate hardware components.

Moreover, even if one were to consider the more detailed disclosure of *Su* (which does ***not*** qualify as prior art), it still does not disclose separate hardware components. As mentioned above, the Examiner has several times acknowledged that *Su* does not disclose separate processors. In the Office Action mailed March 1, 2005 and the Final Office Action mailed September 12, 2005, the Examiner stated:

> ***Su did not specifically disclose said processors being separate as in claims 5, 12, and 35,*** being DSP as in claim 6, 13 and 36.

(p. 6) (emphasis added). While the Examiner later stated in the Final Office that this statement is referring only to the DSP processors, that is plainly not true. The Examiner has stated that, in addition to not disclosing DSP as in Claims 6, 13, and 36, *Su* also does not disclose the "processors being separate as in claims 5, 12, and 35."

Furthermore, *Su* expressly states that the invention is described in terms of "functional block components" which may be implemented using "any number of hardware components or software elements":

> The present invention may be described herein in terms of ***functional block components*** and various processing steps. It should be appreciated that such functional blocks may be realized by ***any number of hardware components or software elements*** configured to perform the specified functions. For

> example, the present invention may employ various integrated circuit components, e.g., memory elements, digital signal processing elements, logic elements, look-up tables, and the like, which may carry out a variety of functions under the control of one or more microprocessors or other control devices.

(Col. 2, ll. 49-59) (emphasis added). *Su* does not specify that the functions of decoders 230 and 234, mixer 238 and 240, and encoder 232 and 236 are assigned to separate processors. Indeed, *Su* provides that the functional blocks may be implemented in software.

Another passage in *Su* further indicates that Figure 2, on which the Examiner relies to support the rejections, is a "simplified schematic" of functional blocks as opposed to hardware components.

> FIG. 2 is a simplified schematic: there might also be certain additional components advantageously coupled between the packet network and the decoders (and encoders). Specifically, with respect to the decoders, there. will likely be **a functional block** (not shown) that receives the packets from packet network 201 and removes all unnecessary routing, encryption, and protection information (a "decapsulator"). Conversely, with respect to the encoders, there will likely be ***a functional block*** (an "encapsulator") for each encoder that receives speech samples from the mixer and adds certain information regarding routing, encryption, and the like prior to sending the packets out over packet network 201.

(Col. 5, ll. 19-31) (emphasis added).

For at least these reasons, *Su* does not disclose, teach, or suggest "the conferencing device comprising two or more processors operable to decode the input data streams to generate the input media information, to mix the input media information to generate output media information, and to encode the output media information to generate output data streams, wherein the processors are separate hardware components," as recited in Claim 40. Because *Su* does not disclose, expressly or inherently, every element of independent Claims 40 and its respective dependent claims, Appellants respectfully request the Board to reverse the Examiner's rejection of independent Claims 40, as well as Claims 32-37 which depend from Claim 40, and direct the Examiner to issue a notice of allowance.

III. THE PROPOSED SU-LEONDIRES COMBINATION FAILS TO DESCRIBE, EXPRESSLY OR INHERENTLY, THE INVENTIONS RECITED IN CLAIMS 5, 6, 12, 13, 35, AND 36.

Appellants' Claim 5 recites:

> The apparatus of Claim 38, wherein the mixing processor and the first media transformation processor are separate integrated circuits.

Claims 12 and 35 recites similar limitations.

Appellants' Claim 6 recites:

> The apparatus of Claim 38, wherein the mixing processor and the first media transformation processor are separate digital signal processors (DSPs).

Claims 13 and 36 recite similar limitations.

The Examiner rejects Claim 5, 6, 12, 13, 35, and 36 under 35 U.S.C. § 103(a) as being unpatentable by Su in view of U.S. Patent No. 5,841,763 ("*Leondires*"). To establish a *prima facie* case of obviousness, there must be a suggestion or motivation in the prior art to modify or combine the references, and the combination of reference must teach or suggest all elements of the rejected claims. *In re Vaeck*, 947 F.2d 488, 20 U.S.P.Q.2d 1438 (Fed. Cir. 1991). The Examiner's rejection of Claim 5, 6, 12, 13, 35, and 36 under 35 U.S.C. § 103 fails both of these requirements. First, even if the combination were proper, the proposed *Su-Leondires* combination fails to teach or suggest all elements of the claims. Second, there is no suggestion or motivation in the cited references or in the prior art to combine *Su* and *Leondires*.

A. Su and Leondires, Whether Taken Alone or in Combination, Fail to Teach or Suggest All Limitations of Claims 5, 6, 12, 13, 35, and 36.

As explained above, Appellants have shown that *Su* fails to disclose all limitations of Claims 38, 39, and 40. Claims 5, 6, 12, 13, 35, and 36, which depend from Claims 38, 39, and 40, includes the limitations that are not taught or suggested by *Su*. *Leondires* fails to remedy the deficiencies of *Su*. For at least this reason, *Su* and *Leondires*, whether taken alone or in combination, fail to teach or suggest all limitations of Claims 5, 6, 12, 13, 35, and 36. Because the references fail to teach all limitations of the claims, Appellants respectfully request the Board to reverse the Examiner's rejection of Claims 5, 6, 12, 13, 35, and 36, and direct the Examiner to issue a notice of allowance.

Furthermore, *Su* and *Leondires* do not disclose, teach, or suggest the particular additional limitations recited in Claims 5, 6, 12, 13, 35, and 36. Here, the Examiner correctly

concludes, "*Su* did not specifically disclose said processors being separate as in claims 5, 12, and 35 [or] being DSP processors as in claims 6, 13, and 36." (Final Office Action at p. 6.) But again, *Leondires* fails to remedy the deficiencies of *Su.* According to the Examiner, *Leondires* "discloses a conferencing device with separate processors." (p. 6). *Leondires*, however, does not disclose, teach, or suggest using separate processors for ***mixing*** and ***encoding***. The portion of the specification cited by the Examiner describes audio decoding digital signal processors (ADPs) and audio encoding digital signal processors (AEPs). The ADPs decode audio information. (Col. 14, ll. 33-43). The AEPs mix ***and*** encode audio information: "The AEPs read the decoded audio signals from DSs time slots, mix the decoded audio signals from each of the conferees and encode the results of the mixing according to the particular G-series standard." (Col. 14, ll. 51-54). Thus, Leondires expressly teaches away from Applicants' claimed invention.

In contrast to the AEPs of *Leondires*, Claims 38 and 39 require two separate processors for mixing and encoding. Claim 39 requires: (1) "a mixing processor operable to mix input media information" and (2) "first media transformation processor operable to receive the output media information from the mixing processor, to encode the output media information to generate an output data stream, and to communicate the output data stream to the second participant's end-user device." Similarly, Claim 39 distinguishes between a mixing processor for mixing and a media transformation processor for encoding. Claim 39 requires the following steps: "mixing input media information associated with two or more first participants to generate output media information for communication to a second participant," "communicating the output media information from a mixing processor to a first media transformation processor," and "encoding the output media information to generate an output data stream."

For the reasons discussed above with respect to independent Claims 38, 39, and 40, as well as these additional reasons, *Su* and *Leondires* do not disclose Applicants' claimed invention recited in dependent Claims 5, 6, 12, 13, 35, and 36. Accordingly, Applicants respectfully request reconsideration and allowance of dependent Claims 5, 6, 12, 13, 35, and 36.

B. There is No Teaching, Suggestion, or Motivation to Combine or Modify the Teachings of Su and Leondires.

Further, the proposed combination of *Su* and *Leondires* is improper because the prior art fails to suggest or motivate the proposed combination of the references. The factual

inquiry whether to combine references must be thorough and searching. *McGinley v. Franklin Sports, Inc.*, 262 F.3d 1339, 1351-52, 60 U.S.P.Q.2d 1001, 1008 (Fed. Cir. 2001). This factual question cannot be resolved on subjective belief and unknown authority, but must be based on objective evidence of record. *See In re Lee*, 277 F.3d 1338, 1343-44, 61 U.S.P.Q.2d 1430, 1434 (Fed. Cir. 2002).

Nothing in *Su* or *Leondires* suggests or motivates the proposed combination. The Examiner, in the *Final Office Action*, merely states that the teachings of one reference would improve the teachings of another reference:

> [I]t would have been obvious to a person of ordinary skill in the art at the time of the invention to provide the DSP means as disclosed in Leondires to the Su system with the motivation of obtaining a conferencing system equipped to service conferees that employ ITU standard wherein the number processing resources can be reduced.

(Final Office Action at p. 6.)

The motivation provided represents the subjective belief of the Examiner, is not substantiated by any known authority, and therefore is not based on objective evidence of record. Thus, the record fails to provide the required evidence of a teaching, suggestion, or motivation to combine or modify the references, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art.

Appellants thus respectfully request the Board to find the proposed *Su-Leondires* combination improper, reverse the Examiner's rejection of Claims 5, 6, 12, 13, 35, and 36, and direct the Examiner to issue a notice of allowance.

CONCLUSION

Appellants have demonstrated that the present invention, as claimed, is clearly distinguishable over the prior art cited by the Examiner. Therefore, Appellants respectfully request the Board of Patent Appeals and Interferences to reverse the Examiner's final rejection of the pending claims and instruct the Examiner to issue a notice of allowance of all pending claims.

The Commissioner is hereby authorized to charge the fee of $500.00 to file this Appeal Brief to Deposit Account No. 02-0384 of Baker Botts L.L.P. The Commission is hereby authorized to charge any additional fee or credit any overpayment to Deposit Account No. 02-0384 of Baker Botts L.L.P.

Respectfully submitted,

BAKER BOTTS L.L.P.
Attorneys for Applicants

Jeffery D. Baxter
Reg. No. 45,560

Correspondence Address:

_X_ Customer Number **05073**

Date: May 10, 2006

Appendix A: Claims Involved in Appeal

1. (Canceled)

2. (Previously Presented) The apparatus of Claim 38, further comprising a second media transformation processor coupled to the mixing processor, the second media transformation processor operable to receive an input data stream from a first participant's end-user device, to decode the input data stream to generate input media information associated with the first participant, and to communicate the input media information associated with the first participant to the mixing processor.

3. (Previously Presented) The apparatus of Claim 38, wherein the first media transformation processor is further operable to receive an input data stream from the second participant's end-user device, to decode the input data stream to generate input media information associated with the second participant, and to communicate the input media information associated with the second participant to the mixing processor.

4. (Previously Presented) The apparatus of Claim 38, wherein the mixing processor is further operable to receive an input data stream from a first participant's end-user device and to decode the input data stream to generate input media information associated with the first participant.

5. (Previously Presented) The apparatus of Claim 38, wherein the mixing processor and the first media transformation processor are separate integrated circuits.

6. (Previously Presented) The apparatus of Claim 38, wherein the mixing processor and the first media transformation processor are separate digital signal processors (DSPs).

7. (Previously Presented) The apparatus of Claim 38, wherein the media conference is a voice telephone conference and the media information is voice information.

8. (Canceled)

9. (Previously Presented) The method of Claim 39, further comprising:

receiving at a second media transformation processor an input data stream from a first participant's end-user device;

decoding the input data stream to generate input media information associated with the first participant; and

communicating the input media information associated with the first participant from the second media transformation processor to the mixing processor.

10. (Previously Presented) The method of Claim 39, further comprising:

receiving at the first media transformation processor an input data stream from the second participant's end-user device;

decoding the input data stream to generate input media information associated with the second participant;

communicating the input media information associated with the second participant from the first media transformation processor to the mixing processor; and

mixing the input media information associated with the second participant with input media information from one or more other participants to generate output media information for communication to a first participant.

11. (Previously Presented) The method of Claim 39, further comprising:

receiving at the mixing processor an input data stream from a first participant's end-user device; and

decoding the input data stream to generate input media information associated with the first participant.

12. (Previously Presented) The method of Claim 39, wherein the mixing processor and the first media transformation processor are separate integrated circuits.

13. (Previously Presented) The method of Claim 39, wherein the mixing processor and the first media transformation processor are separate digital signal processors (DSPs).

14. (Previously Presented) The method of Claim 39, wherein the media conference is a voice telephone conference and the media information is voice information.

15. (Withdrawn) A system resource management (SRM) module coupled to one or more media transformation processors and one or more mixing processors in a conferencing device, the SRM module operable to receive a request to support a media conference and, in response, to allocate the media conference to at least a first media transformation processor and a mixing processor, wherein the mixing processor mixes input media information associated with two or more participants in the media conference to generate output media information and the first media transformation processor encodes the output media information to generate an output data stream for communication to a participant in the media conference.

16. (Withdrawn) The SRM module of Claim 15, wherein:

the SRM module is further operable to communicate to the mixing processor control information identifying the first media transformation processor; and

the mixing processor uses the control information to communicate the output media information to the first media transformation processor.

17. (Withdrawn) The SRM module of Claim 15, wherein the SRM module is further operable to allocate the media conference to a second media transformation processor that decodes an input data stream received from a participant in the media conference to generate input media information.

18. (Withdrawn) The SRM module of Claim 17, wherein:

the SRM module is further operable to communicate to the second media transformation processor control information identifying the mixing processor; and

the second media transformation processor uses the control information to communicate the generated input media information to the mixing processor.

19. (Withdrawn) The SRM module of Claim 15, wherein the SRM module is further operable to store status information identifying the first media transformation processor and mixing processor supporting the media conference.

20. (Withdrawn) The SRM module of Claim 15, wherein the mixing processor and the first media transformation processor are separate integrated circuits.

21. (Withdrawn) The SRM module of Claim 15, wherein the mixing processor and the first media transformation processor are separate digital signal processors (DSPs).

22. (Withdrawn) The SRM module of Claim 15, wherein the media conference is a voice telephone conference and the media information is voice information.

23. (Withdrawn) Media conference migration software embodied in a computer-readable medium in a conferencing device, the conferencing device including one or more media transformation processors and one or more mixing processors, the media conference migration software operable to perform the following steps:

receiving a request to support a media conference;

assigning a mixing processor a task of mixing input media information associated with two or more participants to generate output media information; and

assigning a first media transformation processor a task of encoding the output media information to generate an output data stream for communication to a participant in the media conference.

24. (Withdrawn) The media conference migration software of Claim 23 further operable to perform the step of communicating to the mixing processor control information identifying the first media transformation processor, wherein the mixing processor uses the control information to communicate the output media information to the first media transformation processor.

25. (Withdrawn) The media conference migration software of Claim 23 further operable to perform the step of assigning a second media transformation processor a task of decoding an input data stream received from a participant in the media conference to generate input media information associated with the participant.

26. (Withdrawn) The media conference migration software of Claim 25 further operable to perform the step of communicating to the second media transformation processor control information identify the mixing processor, wherein the second media transformation processor uses the control information to communicate the generated input media information to the mixing processor.

27. (Withdrawn) The media conference migration software of Claim 23 further operable to perform the step of storing status information identifying the tasks assigned to the first media transformation processor and the mixing processor.

28. (Withdrawn) The media conference migration software of Claim 23, wherein the mixing processor and the first media transformation processor are separate integrated circuits.

29. (Withdrawn) The media conference migration software of Claim 23, wherein the mixing processor and the first media transformation processor are separate digital signal processors (DSPs).

30. (Withdrawn) The media conference migration software of Claim 23, wherein the media conference is a voice telephone conference and the media information is voice information.

31. (Canceled)

32. (Previously Presented) The system of Claim 40, wherein the conferencing device further comprises a mixing processor operable to mix the input media information to generate the output media information; and information and one or more media transformation processors operable to encode the output media information to generate the output data streams.

33. (Previously Presented) The system of Claim 40, wherein the conferencing device further comprises one or more media transformation processors operable to decode the input data streams to generate the input media information; and information and a mixing processor operable to mix the input media information to generate the output media information.

34. (Previously Presented) The system of Claim 40, wherein the conferencing device is further operable to identify a coding standard used by a participant's end-user device to encode input media information and to encode output media information for communication to the participant's end-user device using the identified coding standard.

35. (Previously Presented) The system of Claim 40, wherein the processors are separate integrated circuits.

36. (Previously Presented) The system of Claim 40, wherein the processors are separate digital signal processors (DSPs).

37. (Previously Presented) The system of Claim 40, wherein the media conference is a voice telephone conference and the media information is voice information.

38. (Previously Presented) An apparatus for using a plurality of processors to support a media conference, comprising:

a mixing processor operable to mix input media information associated with two or more first participants to generate output media information for communication to a second participant; and

a first media transformation processor coupled to the mixing processor, the first media transformation processor operable to receive the output media information from the mixing processor, to encode the output media information to generate an output data stream, and to communicate the output data stream to the second participant's end-user device,

wherein the mixing processor and the first media transformation processor are separate hardware components.

39. (Previously Presented) A method for using a plurality of processors to support a media conference, comprising:

mixing input media information associated with two or more first participants to generate output media information for communication to a second participant using a mixing processor;

communicating the output media information from the mixing processor to a first media transformation processor, wherein the mixing processor and the first media transformation processor are separate hardware components;

encoding the output media information to generate an output data stream using the first media transformation processor; and

communicating the output data stream from the first media transformation processor to the second participant's end-user device.

40. (Previously Presented) A system for using a plurality of processors to support a media conference, comprising:

a plurality of end-user devices coupled to a data network and operable to generate input media information, to encode the input media information to generate input data streams, and to communicate the input data streams using the data network; and

a conferencing device coupled to the data network, the conferencing device comprising two or more processors operable to decode the input data streams to generate the input media information, to mix the input media information to generate output media information, and to encode the output media information to generate output data streams, wherein the processors are separate hardware components;

wherein the end-user devices are further operable to receive the output data streams and to decode the output data streams to generate output media information.

Appendix B: Evidence

Provision Application No. 09/547,832

("*Su Provisional Application*")

Please type a plus sign (+) inside this box → +

jc658 U.S. PTO

Docket Number:	50944.4600

PROVISIONAL APPLICATION FOR PATENT COVER SHEET (Large Entity)

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53 (c).

jc549 U.S. PTO 60/128873 04/12/99

INVENTOR(S)/APPLICANT(S)

Given Name (first and middle [if any])	Family Name or Surname	Residence (City and either State or Foreign Country)
Eyal	Shlomot	
Huan-Yu	Su	
Adil	Benyassine	
Yang	Gao	

☐ *Additional inventors are being named on page 2 attached hereto*

TITLE OF THE INVENTION (280 characters max)

METHOD AND APPARATUS FOR CONFERENCE BRIDGE PROCESSING OF SPEECH

CORRESPONDENCE ADDRESS

Direct all correspondence to:

☐ Customer Number → *Place Customer Number Bar Code Label here*

OR

☒ Firm *or* Individual Name	Mark M. Takahashi, SNELL & WILMER				
Address	400 E. Van Buren				
Address	One Arizona Center				
City	Phoenix	State	AZ	ZIP	85004
Country	USA	Telephone	602/382-6270	Fax	602/382-6070

ENCLOSED APPLICATION PARTS *(check all that apply)*

☒ Specification — *Number of Pages* 8

☒ Drawing(s) — *Number of Sheets* 4

☐ Other (specify)

METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT *(check one)*

☒ A check or money order is enclosed to cover the filing fees

☐ The Commissioner is hereby authorized to charge filing fees or credit any overpayment to Deposit Account Number:

FILING FEE AMOUNT: $150.00

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

☒ No

☐ Yes, the name of the U S Government agency and the Government contract number are:

Respectfully submitted,

SIGNATURE M M. Tak — DATE April 12, 1999

TYPED or PRINTED NAME Mark M. Takahashi — REGISTRATION NO. *(if appropriate)* 38,631

TELEPHONE 602/382-6270

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

SEND TO:* *Box Provisional Application, Assistant Commissioner for Patents, Washington, DC 20231

PROVISIONAL APPLICATION FOR PATENT COVER SHEET (Large Entity)

INVENTOR(S)/APPLICANT(S)		
Given Name (first and middle [if any])	Family Name or Surname	Residence (city and either State or Foreign Country)

Certificate of Mailing by Express Mail

I certify that this provisional patent application cover sheet, provisional patent application and fee is being deposited on April 12, 1999 with the U.S. Postal Service as "Express Mail Post Office to Addressee" service under 37 C.F.R. 1.10 and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

CKtate

Signature of Person Mailing Correspondence

Claudia Tate

Typed or Printed Name of Person Mailing Correspondence

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

SEND TO: Box Provisional Application, Assistant Commissioner for Patents, Washington, DC 20231

METHOD AND APPARATUS FOR CONFERENCE BRIDGE PROCESSING OF SPEECH

Inventors: **Eyal Shlomot**
Huan-Yu Su
Adil Benyassine
Yang Gao

FIELD OF THE INVENTION

The present invention relates generally to telecommunication systems. In particular, the present invention relates to the processing of speech signals. More particularly, the present invention relates to the processing of speech signals in the context of conference call bridging.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the present invention may be derived by referring to the detailed description when considered in connection with the following Figures:

FIG. 1 is a schematic representation of a conference bridging system in accordance with the present invention;

FIG. 2 is a schematic representation of a conference bridging element in accordance with the present invention;

FIG. 3 is a schematic representation of an exemplary configuration that may be utilized in a practical application; and

FIG. 4 is a flow diagram of an exemplary intelligent bridging process in accordance with the present invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS OF THE INVENTION

The following description of the preferred exemplary embodiments are not meant to limit the scope of the present invention in any way. Those skilled in the art will recognize that changes and modifications may be made to the preferred embodiments without departing from the scope of the present invention. These and other changes or modifications are intended to be included within the scope of the present invention, as broadly described herein.

Intelligent Mixing of Speech Channels for a Conference Bridge

Description of the problem

A conference bridge enables a conference call, as multiple input speech channels are mixed together and then fed into multiple output speech channels. The input speech channels are digital, and carry the speech information in a coded digital form. The digital form for each channel is a bit stream, generated by a speech encoder at the remote end of the conference call. Each bit stream can be generated by a different speech coding standard, for example, G.711, G.726, G.729(A), or G.723.1 (at two possible bit rates). One possible approach for the mixing of the several speech channels is the decoding of the speech (each channel with its appropriate decoder), summation of the speech signal into a single channel, and re-encoding of the mixed channel (with the appropriate encoders) to generate the bit streams for the multiple output channels.
Several problems are encountered with this direct mixing approach. The problems arise for the case of one active talker at a given time, as well as for the case of multiple active talkers at a given time. First, even for a signal active talker, it is clear that in this approach each speech signal is coded twice, first to generate the bit stream into the conference bridge, and then to generate the bit stream out of the conference bridge. It is well known that this tandem coding result in a degradation of the speech. Another problem arises when several talkers try to talk at the same time. Since low-bit rate speech coders are highly tuned for a single talker (by using, for example, a limited order spectral model and a single pitch representation), they are unsuitable for the coding of a signal that is comprised of several talkers at the same time. Another issue is the computation complexity in the conference bridge. While several speech parameters, such as spectrum, pitch, energy, level of background noise, are known for each individual decoder, they have to be re-computed by the encoder of the mixed signal.

Our solution

We propose the approach of intelligent conference bridge operation. Intelligent bridge comprises of 4 basic steps. At the first step all input speech channels are aligned and a common framing is established, and parameters extraction is performed for channels that use non-parametric coders. The second step involves an intelligent speech mixing of the speech waveform of the input channels, the third step is an intelligent mixing of the parameters of the input speech channels, and the fourth step is an intelligent re-encoding of the mixed output speech channels. These steps can incorporate priority assignment and speech enhancement (for example, by noise reduction or reshaping) for each input and output channel.
This second step and third steps require the modification of the standard speech decoders for their special operation in the conference bridge, and the third and fourth step require the modification of the standard speech encoders for their special operation in the conference bridge.

Framing and Alignment for Speech Mixing in a Conference Bridge

Description of the problem

Several coded speech channels are the input into the conference bridge. The speech at each channel is represented by a bit stream of a speech coding scheme. Not only the format of the bit stream is different from one coding scheme to the other, but also the frame size, for example, from 30 ms in G.723.1 to 20 ms in the futuristic G.4k, to 10 ms in G.729, and to 5 ms for G.728. Moreover, the input bit stream can be coming from a frame-less speech coding approach, such as G.726 or G.711. Intelligent mixing of the speech requires a common frame for the mixing of the parameters.

Our solution

We propose, as a first step in intelligent operation of a conference bridge, the creation of a 'super frame', which is the largest size frame of all of the coding schemes of the speech input channels. For example, if at least one input channel uses the G.723.1 coder, the size of the super frame will be 30 ms. We propose the alignment and the buffering of the short length frames to create a super frame (for example, three 10 ms frames of G.729 to generate a 30 ms super frame suitable for intelligent mixing with G.723.1). We propose the interpolation of the speech parameters from the aligned short length frames to the long length frames, and from the long length frame to the aligned short length frames. We propose creating an aligned super frame structure for the frame-less coding schemes (such as G.711, G.726). We propose parameter extraction and interpolation approach for the non-parametric coders (such as G.711, G.726, and G.728), and the use of these parameters in the intelligent mixing of these coders with other coders.

'Returned-Echo' Cancellation Using Multiple Intelligent Mixing in a Conference Bridge

Description of the problem

A conference call involves several participants. For each participant, the mixed speech information from all the other participants should be provided. One possible solution is the (intelligent) mixing of *all* the channels into a single channel, which is used as the input for each of the output encoders in the conference bridge. The main problem with this approach is that each participant will hear his or her speech, in addition to the speech generated by the other participants. Hearing the speech of oneself, delayed by the two-way digital link and the conference bridge processing time, is perceived as a very annoying returned echo. For an IP based conference bridge, the delay can be of the order of several hundred ms, and the returned echo would be intolerable.

Our solution

We propose the intelligent 'returned-echo' cancellation in a conference bridge. We propose to generate a multiple of mixed signals at the conference bridge, each mixed composed of all the input speech channels, *excluding* the speech of one channel. The mixed signal without the contribution of a particular participant is used as the output speech channel for that particular participant. This mixing scheme removes the contribution of each participant from the signal that is sent back to him/her by the conference bridge, and removes completely the returned echo effect.

Intelligent Spectral Mixing in a Conference Bridge

Description of the problem

The speech spectrum is an important parameter for parametric speech coding. The speech spectrum is commonly represented by the linear prediction (LP) parameters, or by one of their alternative representation, such as normalized autocorrelation function, the reflection coefficients, the arc-sin parameters, the log-area ratios, the line spectral frequencies, the cosines of the line spectral frequencies, as well as the impulse response of the LP filter. Any parametric coder, such as G.723.1 and G.729, transmits a coded representation of the spectrum. It is well known that an accurate representation of the spectrum is crucial for high quality speech, and that the reevaluation of the spectrum is a major source of degradation in tandem coding of speech.

Our solution

We propose to intelligent spectral mixing for the conference bridge. The intelligent spectral mixing uses the decoded spectral information from the mulitple input channels, instead of reevaluating the spectrum of the mixed signal. The spectra can be mixed to provide a meaningful spectral information to the output speech encoder. The spectral mixing can take into consideration the alignment, the framing, the content of each speech input (for example, its energy), as well as timing information, such as a the information about a 'cutting in' talker. The spectral mixing can also be preset to favor specific talker or talkers, providing them a better control over the conference call. The spectral mixing can be performed using any of the representation for the spectrum, described above. In particular, we suggest spectral mixing using the line spectral frequencies (or the cosines of the line spectral frequencies), which are readily available in most parametric coders, in order to reduce the complexity of the conference bridge. We also suggest to obtain a spectral estimate for non-parametric coders, such as G 711, G.726, and G.728, and to use this spectral estimate for the intelligent mixing with parametric coders, such as G.729 and G.723.1.

Intelligent Pitch Mixing in a Conference Bridge

Description of the problem

The pitch is an important parameter in parametric coding of speech. Reevaluating of the pitch from the mixed signal is a simple approach of pitch determination for the mixed output signal in a conference bridge. However, when several participants talk at the same time, the evaluated pitch value might not be meaningful and the mixed signal will be distorted. Moreover, the reevaluation of the pitch will require additional computation for the output channel encoders.

Our solution

We propose an intelligent pitch mixing for the conference bridge. Since each parametric coder, such as G.729 and G.723.1, transmits a description of the pitch we propose to use this pitch information to select a single pitch for the output channel encoders at each time. We propose the mixing of the pitch based on the input channels speech and timing information. We propose this pitch mixing as either a final pitch to be used by the channel output encoders, or as an initial pitch estimate for the channel output encoders. In particular, we propose the selection of a single pitch, based on the energy of the input speech channels and the pitch prediction gain, to be used as an initial estimate for the closed-loop pitch selection, common in low bit-rate coders such as G.723.1 and G.729. We also suggest to obtain a pitch estimate for non-parametric coders, such as G.711, G.726, and G.728, and to use this pitch estimate for the intelligent mixing with parametric coders, such as G.729 and G.723.1.

Priority Assignment in a Conference Bridge

Description of the problem

In a common conference call, the speech signals of all participants are mixed without any priority or preference of one or more participants over the others. Intelligent mixing of speech enables the assignment of a higher or lower priority to one or more participants, which can serve as a tool for managing and controlling the conference call.

Our solution

We propose to use a priority assigning algorithm in intelligent mixing of speech for a conference bridge. A higher or lower priority of a talker can be implemented by a higher or lower weight in mixing his/her speech parameters during parameters mixing, or by a higher or lower level of mixing of the talker speech waveform during the waveform mixing.

__Background Noise Handling for a Conference Bridge__

Description of the problem

Background noise poses a special problem for low bit-rate speech coders, which are incapable of producing a perceptually faithful representation of most types of background noise. As more and more phone calls are placed from mobile phones, this problem becomes more acute in modern telephony systems. It is well known that the representation of the background noise is worse in tandem coding of speech. In a conference bridge, the representation of the background noise is even more important, since several sources of background noise can be mixed together into a single channel, therefore reducing the overall signal-to-noise ratio.

Our solution

We propose a special approach for background noise handling in a conference bridge. We propose tracking the speech and the background noise activity, the background noise level, and the background noise statistics, for each of the incoming channels. We propose modifying the conference bridge speech decoders and the conference bridge speech encoders to enhance the background noise mixing and representation. We also propose to apply a speech enhancement (for example, by noise reduction) for the input speech channels and/or for the combined mixed waveform, to reduce the particular noise from each channel and the overall noise contribution in the conference bridge.

Conference Bridging Product

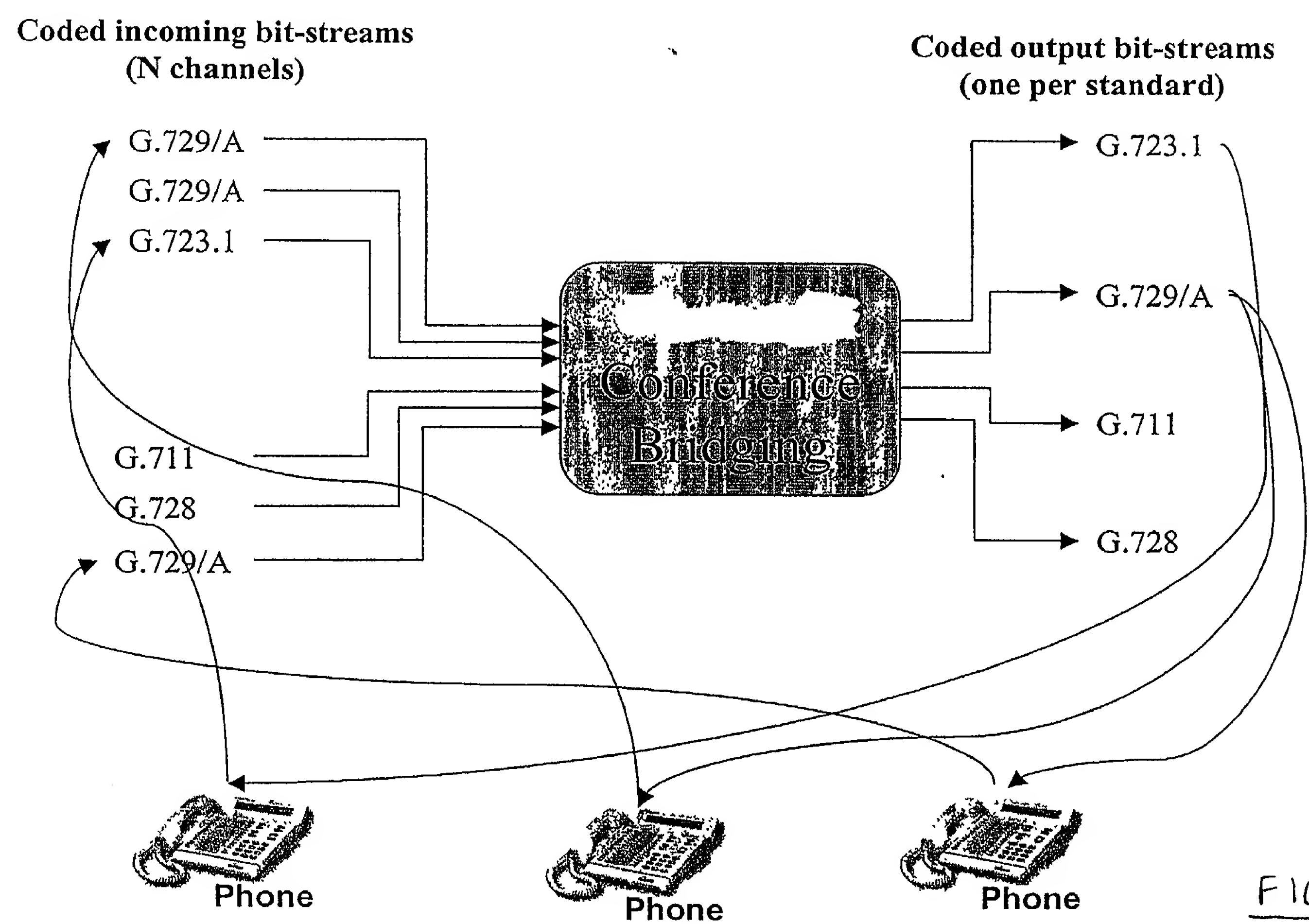
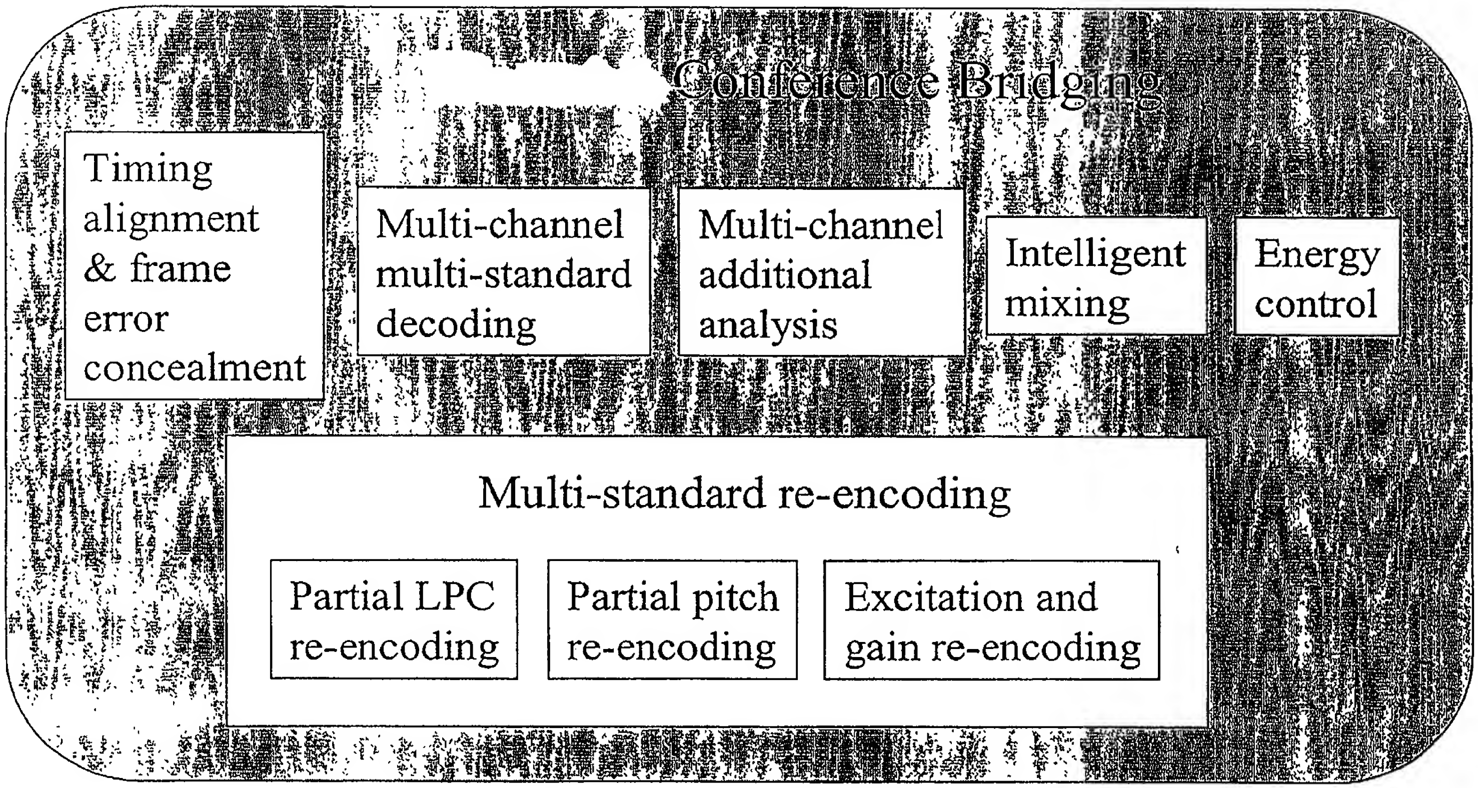


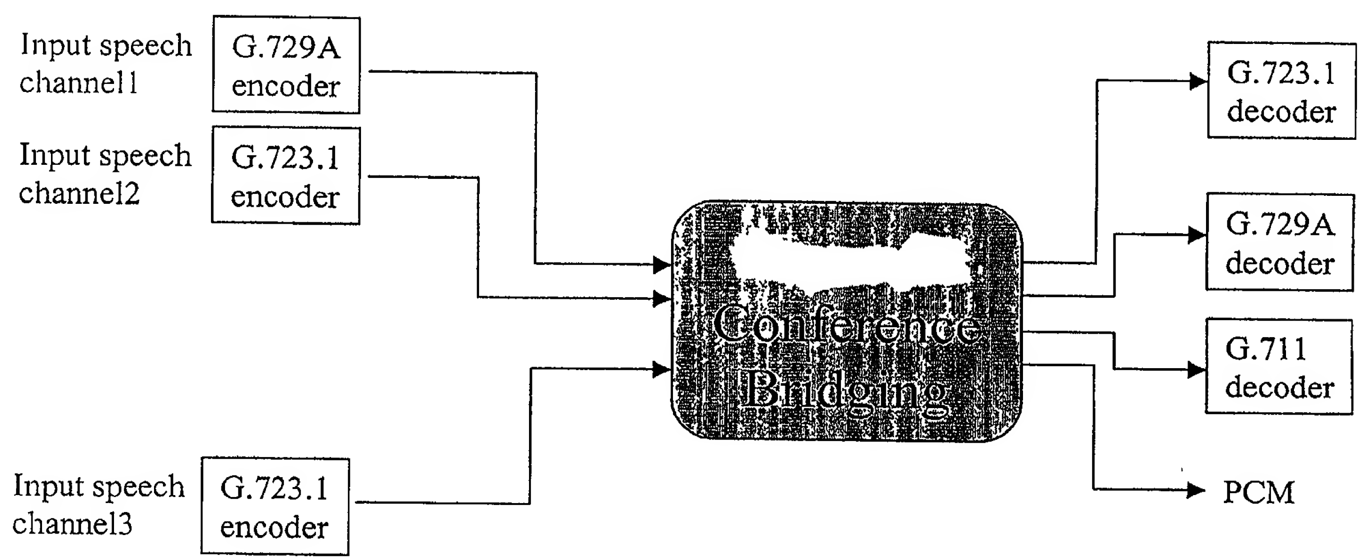
FIG. 1



Conference Bridging
Timing alignment & frame error concealment
Multi-channel multi-standard decoding
Multi-channel additional analysis
Intelligent mixing
Energy control
Multi-standard re-encoding
Partial LPC re-encoding
Partial pitch re-encoding
Excitation and gain re-encoding

FIG. 2

Demo configuration



Notes:

1) PCM is the mixed signal without re-encoding which represents the best possibly achievable quality by the conference bridging.

2) G.729A is used instead of G.729

3) The lower rate of G.723.1 (5.3k) is used in the demo.

FIG. 3

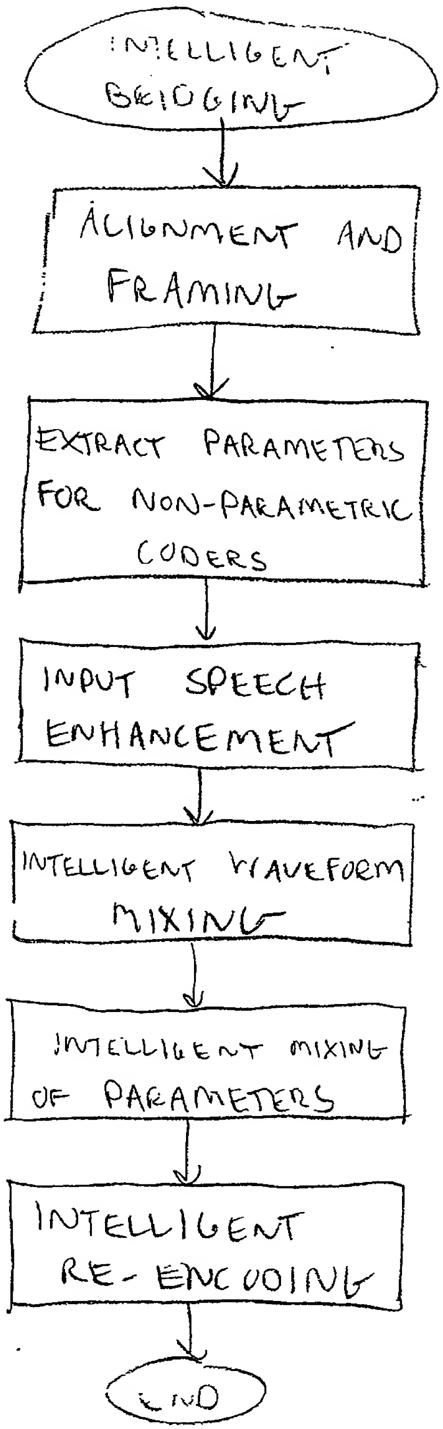

INTELLIGENT BRIDGING
ALIGNMENT AND FRAMING
EXTRACT PARAMETERS FOR NON-PARAMETRIC CODERS
INPUT SPEECH ENHANCEMENT
INTELLIGENT WAVEFORM MIXING
INTELLIGENT MIXING OF PARAMETERS
INTELLIGENT RE-ENCODING
END

FIG. 4

Appendix C: Related Proceedings

NONE